

Design of a Reconfigurable Framework on Instant Information Query for advanced Telematics Services

**Shing Tenqchen^{2&3}, Yung-Kuei Huang¹, Hsin-Hsun Huang¹, Foun-Shea Chang¹,
Kluo-Yueh Chen¹, Saad Laraqui³**

¹The Institute of Ministry of Transportation and Communication, R.O.C.

²ChungHwa Telecom Labs., 12, Lane 551, SEC 5, Min-Tsu RD, Yang-Mei, Taoyuan, Taiwan 326

³Graduate School of Management, UMUC, 3501, University Blvd, East, Adelphi, MD 20783, USA

E-mail: stc@cht.com.tw & klyuen@iot.gov.tw

ABSTRACT

In this paper, we propose an integrated design of a reconfigurable test bed on instant information query framework for advanced Telematics Service with bundle services. We developed a .NET Framework for open Telematics system based on the open gateway which is interoperable with other Telematics systems under heterogeneous wireless network such as GPRS (2.5G), HSPA (3.5G), and WiMAX (3.5G). We provide at least 18 bundle services together for subscribed members. For instant information, we use the unified interface of traffic info center to obtain the real-time traffic info with broadcasting for every five minutes update. The OBU equipments can be cellular phone and OBU on vehicle use PUSH button to show ID number, Mobile number, GPS coordinate, request for service type, status report (accident, construction, traffic jam ...). The core functions of integrated bundle services can include four major services such as instant traffic info, driving info, information safety, and fleet management information, etc. Each of these modules has at least 6 to 9 minor services.

1. INTRODUCTION

Telematics systems nowadays become more important on the intelligent transportation system (ITS) ([1], [3], [5]-[6]) because it offers a two-way communications between human, vehicle, and roads. The instant information is very important for everyone to make a correct decision under the changing world. Especially, the instant traffic information is important to everyone when you driving a vehicle on the road. It will save your time when you know the traffic jam in front and you know the other way to avoid.

The old-fashion Telematics is a standalone, without wireless communications, and vehicle centric services. The next-generation Telematics based on the newly developed technologies will provide a distributed connected multiple wireless communication and driver/passenger centric services ([8]-[10]). In broad sense: services delivered to mobile users with a specific focus on, or a particular added value in an automobile environment. In narrow sense, on-board units can communicate with road-side units or remote servers outside the vehicle.

Thus, the Telematics driving forces can be regarded as they come from car manufacturers, government (ITS), and Telecom operators. The Telematics OBU can be designed to provide services from communications and information technology (IT) industries. Its main function creates location-based and mobile multimedia services, which include remote diagnostics, car safety, navigation, personal information management (PIM), electronic toll collection (ETC), emergence call, etc. The location based services, vehicle positioning system (VPS), advanced transportation information system (ATIS), fleet management, m-commerce, entertainment, etc. The most important factor is the design of service platform as shown in Figure 1([10]).

The main problem of this paper is to provide a reconfigurable framework on instant information query for advanced Telematics services. The Telematics services on the different OBU at the client side are primarily to combine the convergence of information's services, service platform, outside vehicle services, inside vehicle services, and call center to provide an integrated service system with firewall security protected realized architecture as shown in Figure 2.

The rest of paper is organized as following. Section II will discuss the reconfigurable framework and instant information query. Since the framework design is very important to provide Telematics services for different output devices like OBU and cellular phone, the design of integrated structure of Telematics service and the OSGi-like framework and remote management architecture for OBU and cellular phone will be discussed on Section III. In

section IV, we will discuss the system architecture of information management system with statistical records. Then, some of the provided services in the case of driving scenarios will be presented to verify and validate our idea. Finally, we will come out a short conclusion.

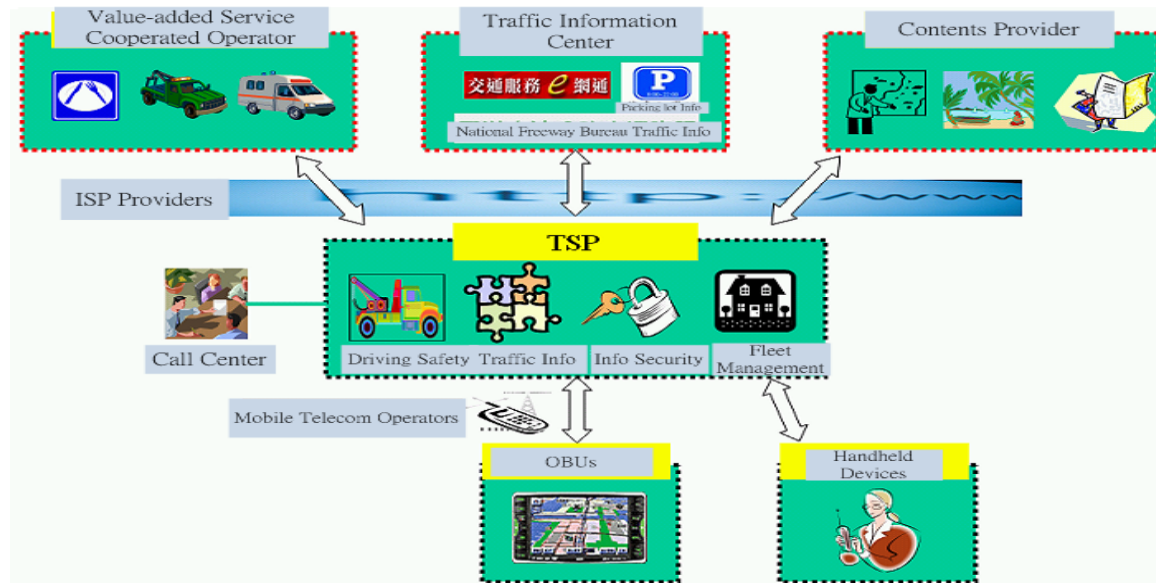


Fig. 1 The main Telematics Service platform

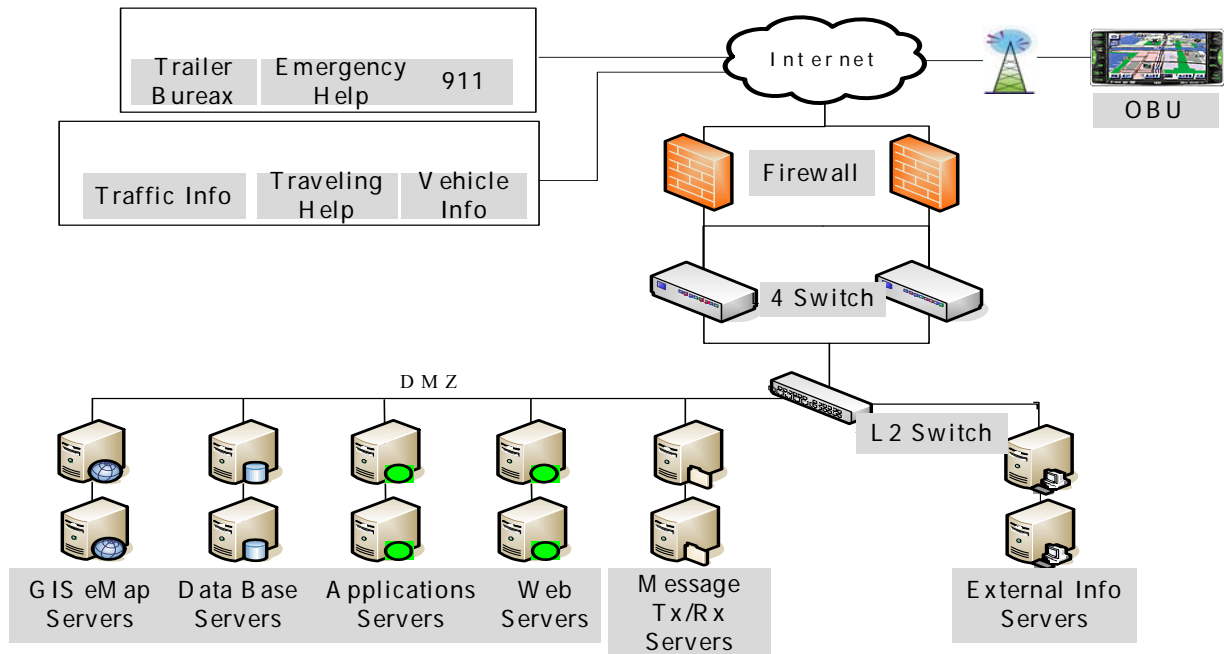


Fig. 2 A complete realized Integrated Telematics Service

II. THE RECONFIGURABLE FRAMEWORK AND INSTANT INFORMATION QUERY

A: Proposed Architecture

The Telematics service is suggested to follow the Microsoft .NET framework, which is an alternative scope of OSGi (Open Service Gateway initiative) as shown in Fig.3. The .NET Framework offers a number of advantages to developers. The webhosting net gives trustworthy, consistent, scalable and secure web hosting services to both individuals and companies. Web hosting service offers a server to host sites, domain registration, build up basic infrastructure. Server automatic online backup is taken as frequently as possible with the help of well

designed backup software. Companies want low cost web hosting provider having no pretensions that the cheap web hosting providers will give 24/7 customer service. The cheap web hosting providers balance their expenses via lower customer service like low quality web design sites and so on.

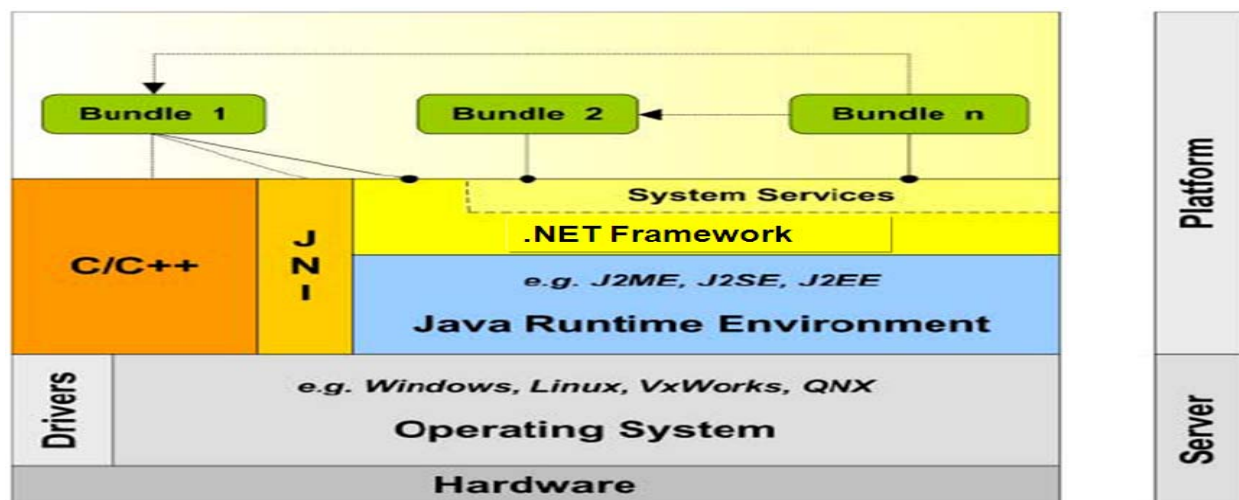


Fig.3 .NET Framework with Service gateway & Bundle Services

The procedure of .NET Framework with service bundle is described as following. The bundle files (native JAR) are installed with bundles to become a package with Lookup table. The user registered to become a member can activate any service from lookup table in http server to enjoy the provided instant contents as shown in Figure 4.

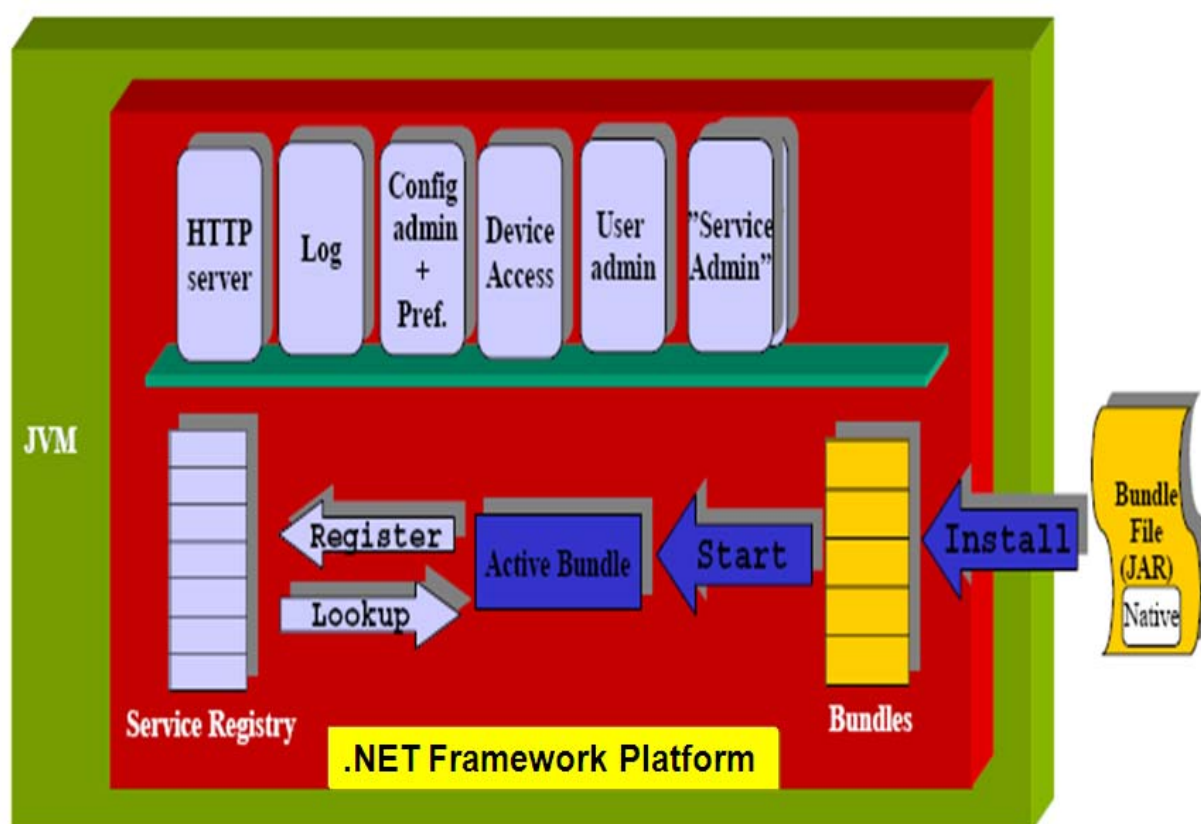


Fig. 4 .NET Framework with Bundle Service

B: Instant Information Query

The procedure of instant information query for user requesting a service from cellular phone or OBU shall include

three basic steps: step 1: verify dependencies; step 2: register service; and step 3: request a service as shown in Figure 5 to Figure 7. Step 1 to step 3 activates the internal procedure from the service event under framework. One can see the legend from the different colors.

Step: 1: Verify Dependencies

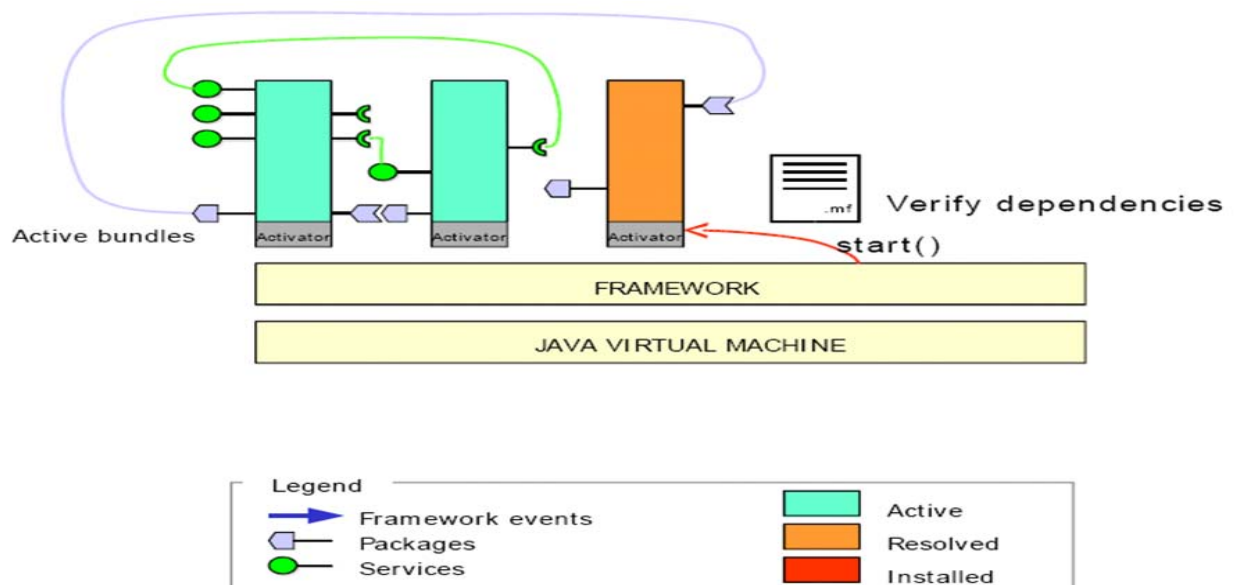


Fig. 5 Verify dependencies during request (source: YCH (2009) [11])

Step 2: Register Service

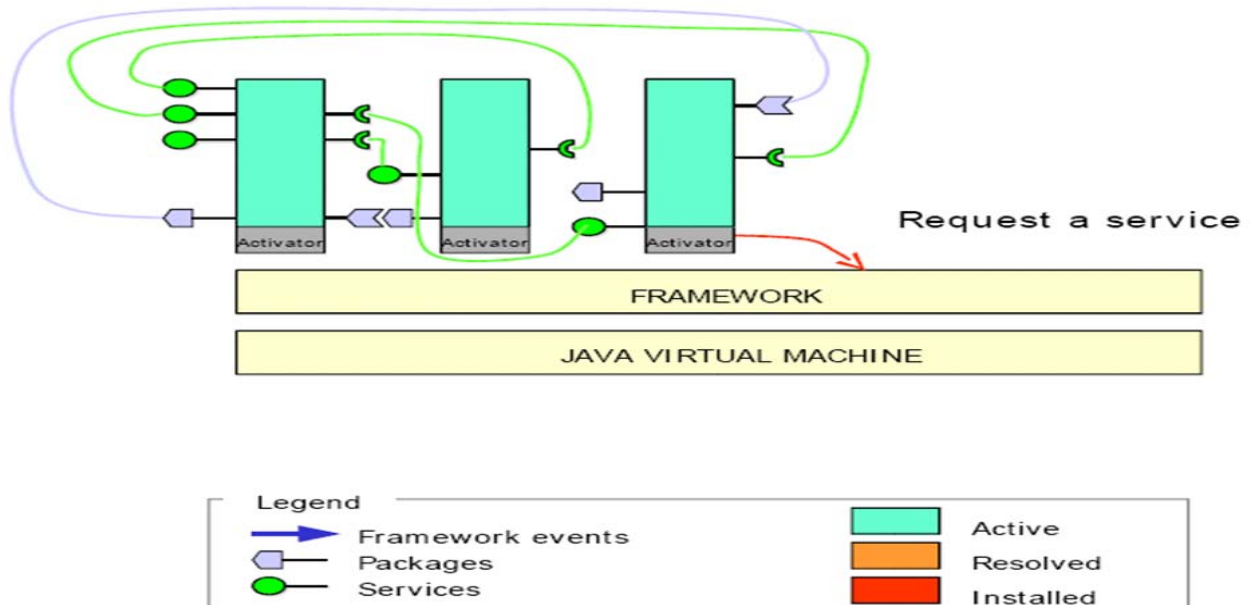


Fig. 6 Register Service (source: YCH (2009) [11])

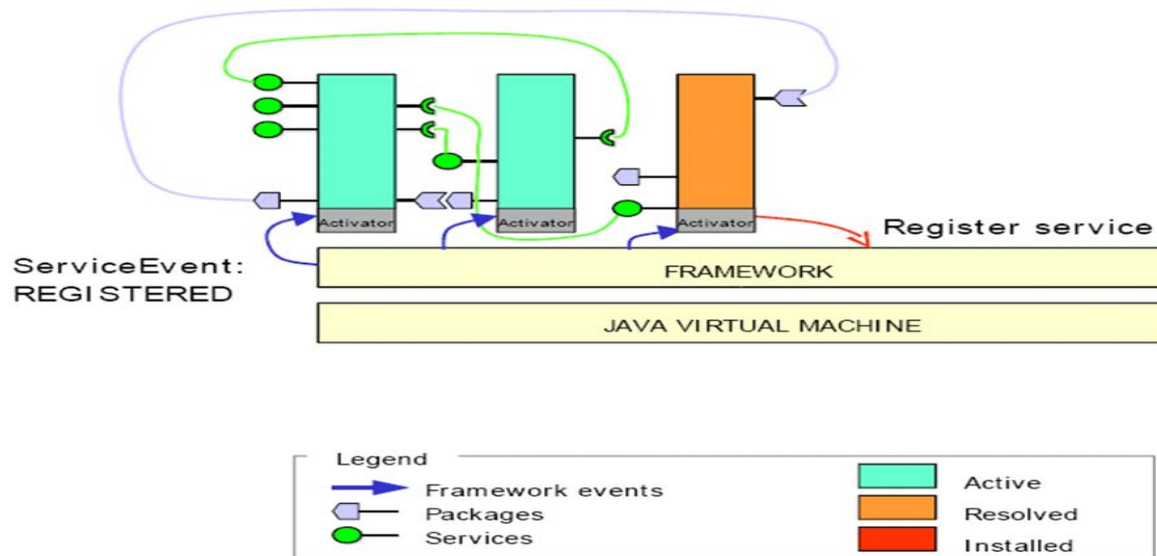


Fig. 7 Request Service (source: YCH (2009) [11])

3. The OSGi-like Framework and Remote Management Architecture

The following paragraphs describe the OSGi-like .NET Framework and remote management architecture for wireless communications. The .NET framework has the consistent programming model in software. One can see the several benefits of .NET Framework in detail.

(a). Consistent Programming Model

Different programming languages have different approaches for doing a task. For example, accessing data with a Virtual Basic (VB) 6.0 application and a VC++ application is totally different. When using different programming languages to do a task, a disparity exists among the approach developers use to perform the task. The difference in techniques comes from how different languages interact with the underlying system that applications rely on.

With .NET, for example, accessing data with a VB .NET and a C# .NET looks very similar apart from slight syntactical differences. Both the programs need to import the system.data namespace, both the programs establish a connection with the database and both the programs run a query and display the data on a data grid. The VB 6.0 and VC++ example mentioned in the first paragraph explains that there is more than one way to do a particular task within the same language.

The functionality that the .NET Class Library provides is available to all .NET languages resulting in a consistent object model regardless of the programming language the developer uses.

(b). Direct Support for Security

Developing an application that resides on a local machine and uses local resources is easy. In this scenario, security isn't an issue as all the resources are available and accessed locally. Consider an application that accesses data on a remote machine or has to perform a privileged task on behalf of a non-privileged user. In this scenario security is much more important as the application is accessing data from a remote machine.

With .NET, the Framework enables the developer and the system administrator to specify method level security. It uses industry-standard protocols such as TCP/IP, XML, SOAP and HTTP to facilitate distributed application communications. This makes distributed computing more secure because .NET developers cooperate with network security devices instead of working around their security limitations.

(c). Simplified Development Efforts

Let's take a look at this with Web applications. With classic ASP, when a developer needs to present data from a database in a Web page, he is required to write the application logic (code) and presentation logic (design) in the same file. He was required to mix the ASP code with the HTML code to get the desired result.

ASP.NET and the .NET Framework simplify development by separating the application logic and presentation logic making it easier to maintain the code. You write the design code (presentation logic) and the actual code (application logic) separately eliminating the need to mix HTML code with ASP code. ASP.NET can also handle the details of maintaining the state of the controls, such as contents in a textbox, between calls to the same ASP.NET

page.

Another advantage of creating applications is debugging. Visual Studio .NET and other third party providers provide several debugging tools that simplify application development. The .NET Framework simplifies debugging with support for Runtime diagnostics. Runtime diagnostics helps you to track down bugs and also helps you to determine how well an application performs. The .NET Framework provides three types of Runtime diagnostics: Event Logging, Performance Counters and Tracing.

(d). Easy Application Deployment and Maintenance

The .NET Framework makes it easy to deploy applications. In the most common form, to install an application, all you need to do is copy the application along with the components it requires into a directory on the target computer. The .NET Framework handles the details of locating and loading the components an application needs, even if several versions of the same application exist on the target computer. The .NET Framework ensures that all the components the application depends on are available on the computer before the application begins to execute.

The mobile network providers suggest regardless to what the service providers want it or not. The core part of .NET framework specifications is a framework that defines an application life cycle management model, a service registry, an Execution environment and Modules. Based on this framework, a large number of OSGi Layers, APIs, and Services have been defined for our applications and is also a standard of GTP.

Applications or components (coming in the form of bundles for deployment) can be remotely installed, started, stopped, updated and uninstalled without requiring a reboot for users. The management of Java packages/classes is specified in great detail. Life cycle management is done via APIs which allow for remote downloading of management policies. The service registry allows bundles to detect the addition of new services, or the removal of services, and adapt accordingly.

(e) Unifies Different Programming Models

The Common Language Runtime has a mapping of data types for programming language to map from framework. The Just-in-time (JIT) compilers compile intermediary language (MSIL) into native code and highly optimized for platform or device as shown in Figure 2. The unified different programming models is designed to consistent Application identifier (API) shown in Figure 8.

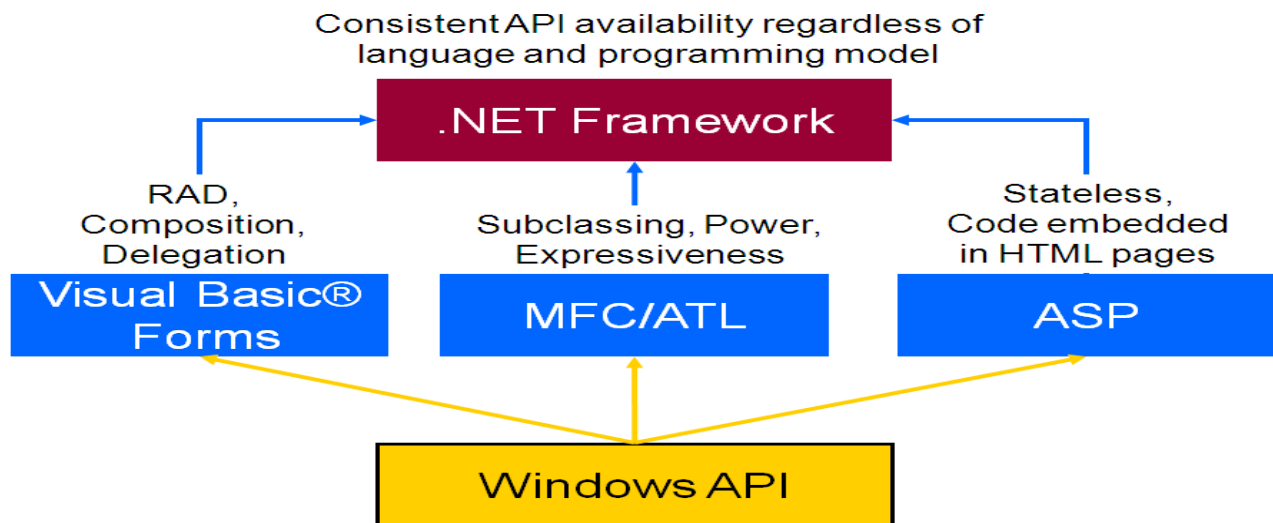


Fig. 8 Unifies Different Programming Models (source: .NET Framework [4])

(f) Orchestration of different applications

The orchestration of different applications is important. It can create a solution through integration of disparate services and applications, multiple data streams, and servers. Thus, what we do is the unify data and messages from everywhere. It can save time by using existing infrastructure and solutions and integrate legacy solutions with .NET solutions. The orchestration of different applications is shown in Figure 9.

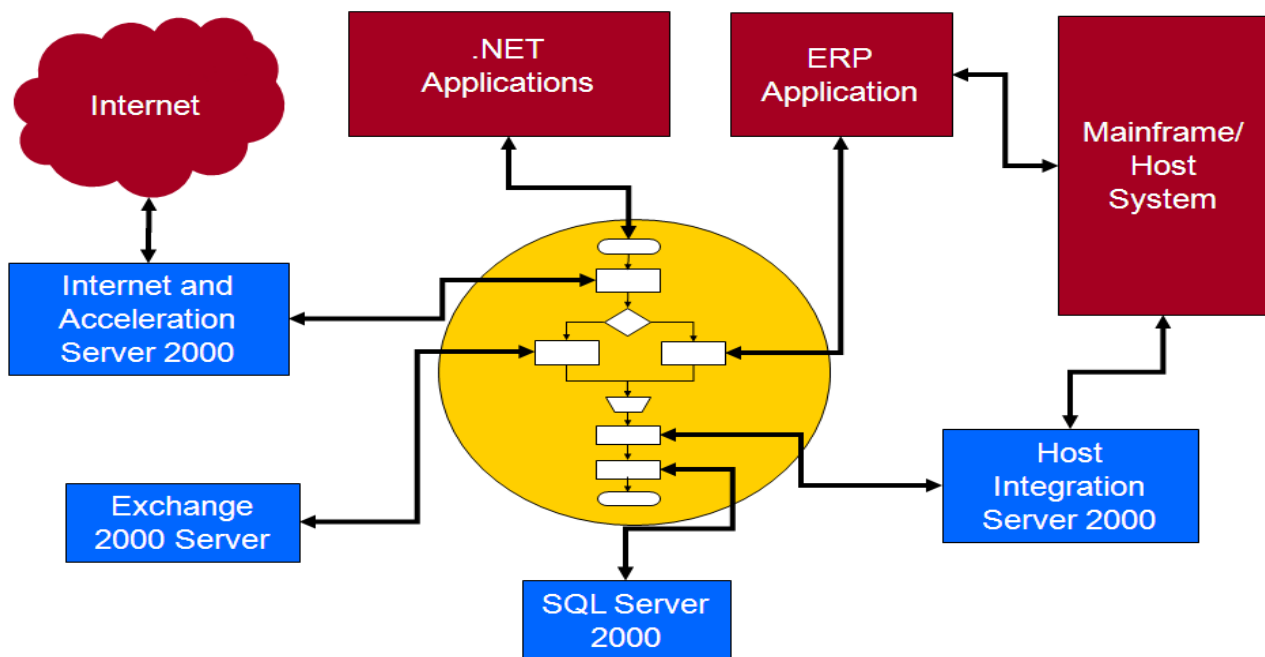


Fig. 9 The orchestration of different applications (source: .NET Framework [4])

The OBU also includes a block to handle services portal is shown in Fig. 10. This portal hosts advanced value-added services and complex speech-recognition services and will allow the provision of an open service access interface for third-party developments like instant traffic information.

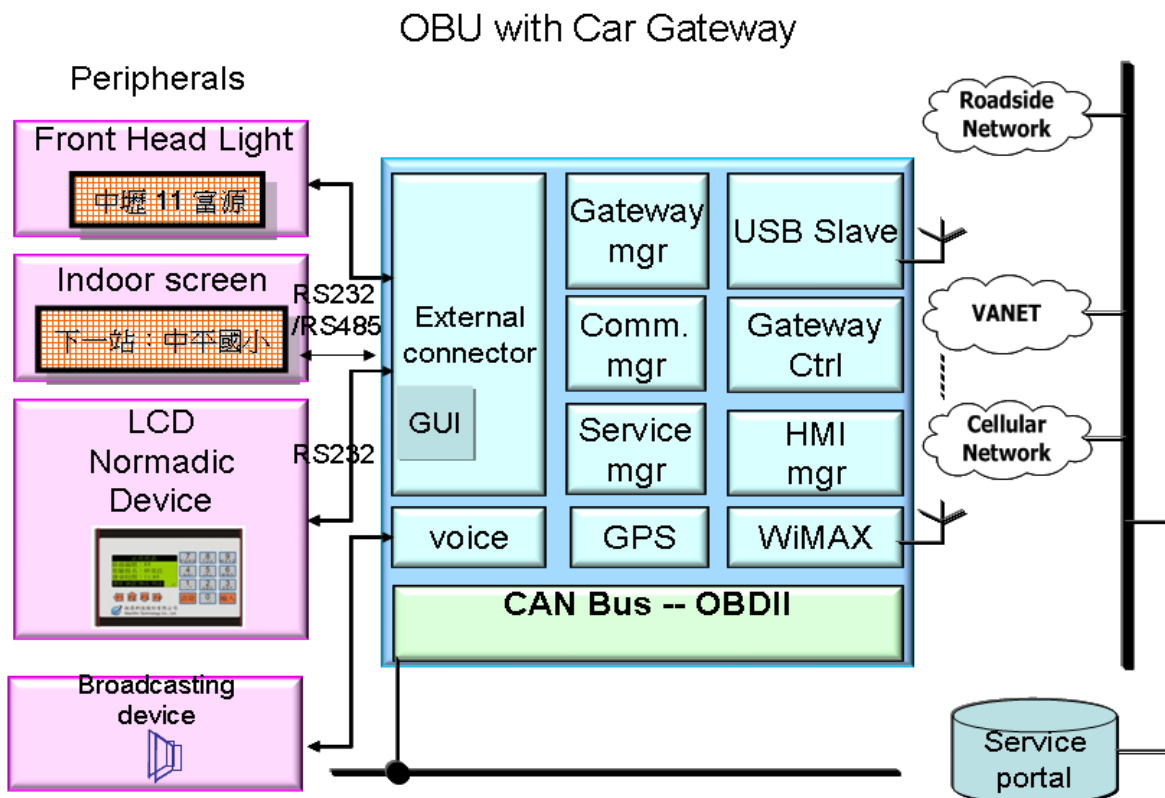


Fig. 10 A reconfigurable Test bed of Integrated Telematics Service

The reconfigurable test bed of integrated Telematics Service is designed to have bundle services and typical remote management architecture. According to the OSGi evolution and contents from R1 2000 to R4 2005, the core specification (spec) of Service Manager shall handle the content framework implementer's spec and bundle programmer's guide as shown in Figure 11. The normative and informative contents are provided according to the requests of registered users.

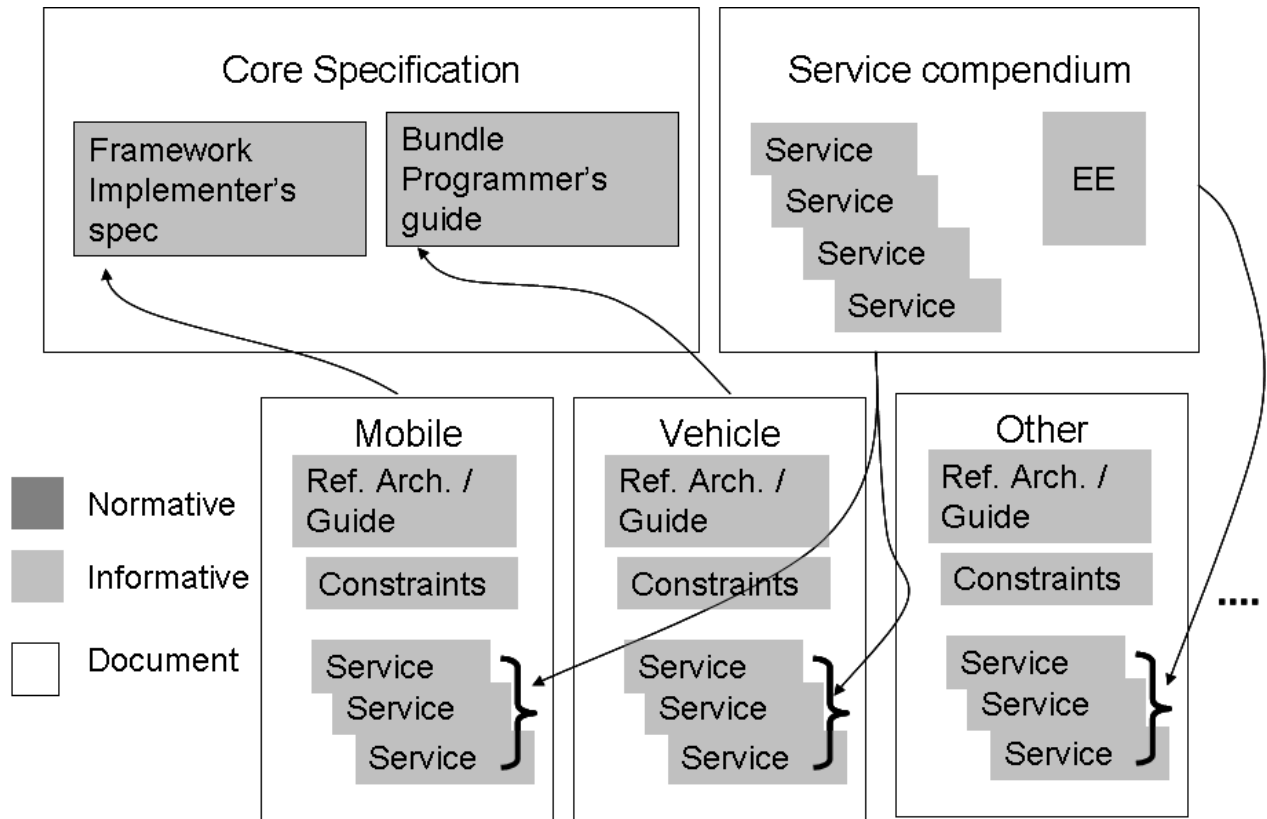


Fig. 11 the OSGi-like .NET Service Manager

The ongoing work of typical remote management architecture will handle the following tasks in the following areas:

- Remote Management – The .NET Service Platform has been carefully crafted to be diagnostic of the management protocol. However, the market might require a preferred standardized management protocol to minimize the number of options and improve inter-operability.
- Power Management – Mobile devices require careful management of the power consumption. Today, many peripherals provide the capability to reduce their power consumption without completely losing their context. The OSGi Alliance is investigating standardizing the interfaces to enable optimized power management.
- Web Services – Amazon, Google, Microsoft.NET, and many more Web services are becoming more popular each day. The .NET Service Platform is an excellent platform for Web Services. Its superb dynamic update facilities the extension of the functions of applications. The rich software environment is designed for Java software, and cooperative facilities that the .NET Service Platform offers are an ideal combination with Web Services.
- Application Servers – The .NET Service Platform is an excellent Java application server. Research is being performed to decide if the .NET Framework is applicable in J2EE environments. The .NET Service Platform's superior life-cycle management architecture could add significantly easier remote management to J2EE servers.
- Connectivity – Embedded applications are more and more confronted with portable devices. iPods, mobile phones, PDAs are just a few examples of devices that people expect to collaborate with their car and home computers. Managing this complexity is one of the main goals of the OSGi Alliance.
- Distribution – The current specifications assume a single Virtual Machine (VM) for the service collaboration

model. There is an increasing interest in using the service model for distribution.

- Native code – Java has many advantages over other environments but many systems require integration with legacy code or code that has special requirements. Power Management – Mobile devices require careful management of the power consumption. Today, many peripherals provide the capability to reduce their power consumption without completely losing their context. The OSGi Alliance is investigating standardizing the interfaces to enable optimized power management.

Using the interfaces provided by the software and hardware platform described as above, the hybrid mode will be able to monitor the quality of all available links to a given destination at any moment. Quality parameters may range from loss of coverage or availability to transmission parameters such as throughput, packet loss rate to signal-to-noise ratio. Based on this monitoring, one can see the results from backend system computer.

4. Mobile Office Service SYSTEM FOR different Client OBUs with Bundle Services

The system architecture and human interface (HMI) service module will be classified into value-added service operators, traffic information center, content service providers, etc, via the communicational interface of Telematics Service Providers (TSP).

(1) ITS/TSP Platform and Statistical Reports

The common platform to provide the wireless communications for different custom peripheral equipment (CPE) such as OBU installed on cars, cellular phone, UMPC, smart phone/PDA, etc, as shown in Figure 12. All of the services platform will provide an instant traffic info, fleet management, secure info, parking info, via the call-center, machine-to-machine, and man-to-man, etc.

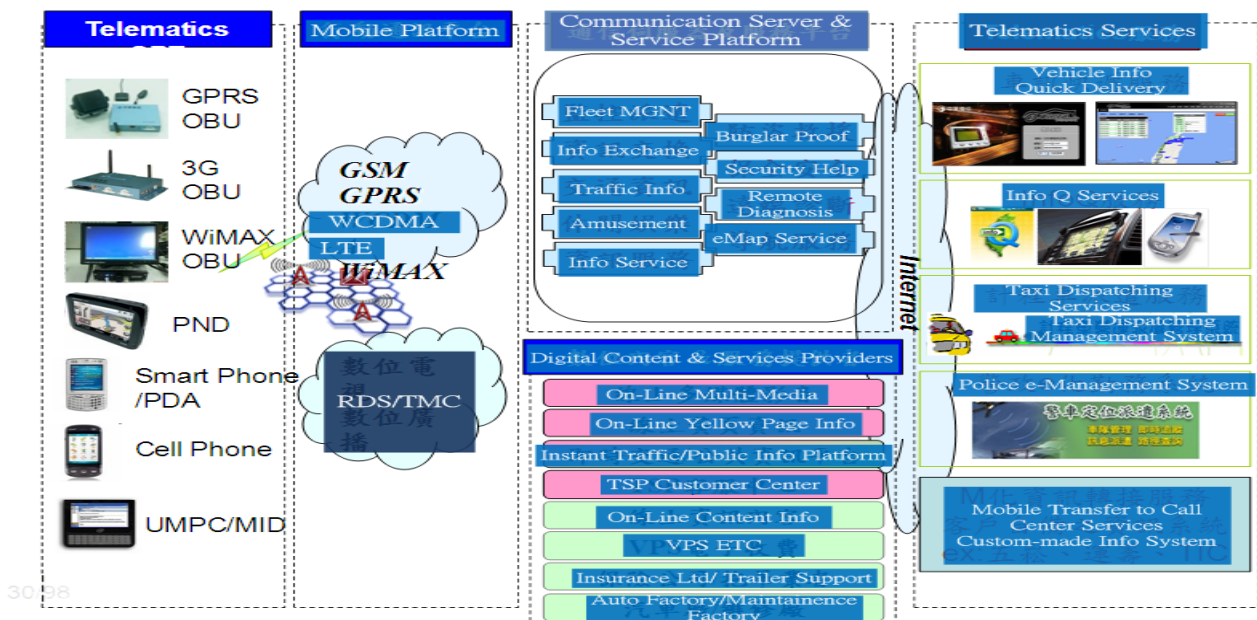


Fig. 12 ITS/Telematics Service Platform

The yearly, monthly, and daily statistical reports of Info Q are given in the following Figures 13 (a), (b), & (c).



Fig. 13 (a) Yearly Statistical Report of Info Q services

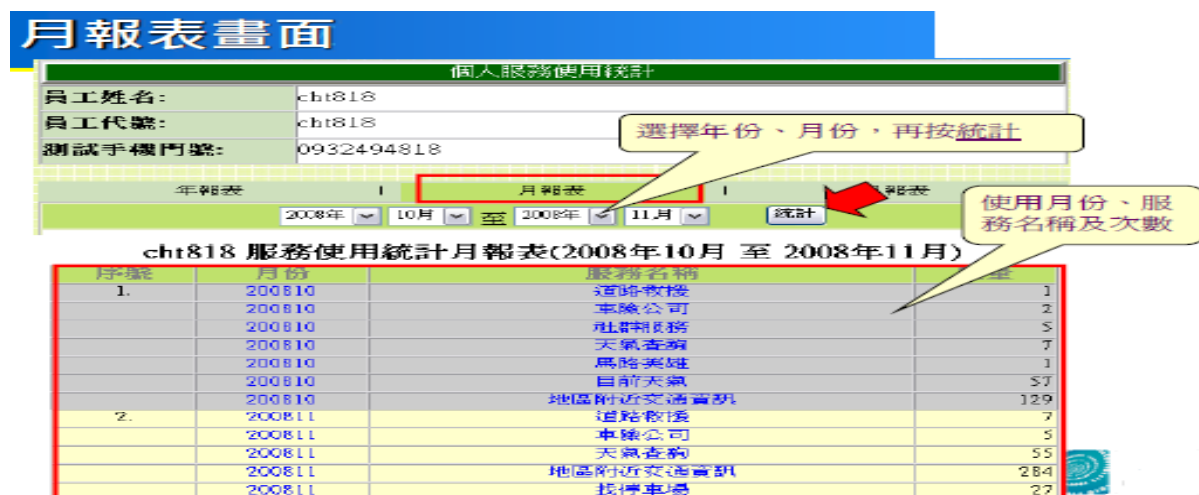


Fig. 13 (b) Monthly Statistical Report of Info Q services



Fig. 13 (c) Daily Monthly Statistical Report of Info Q services

(2) Four major services style to fit the scenarios of driving conditions

We have four major kinds of service style to fit the driving needs. Driver can use the Info Query services at OBU as shown in Figure 14. The Info Q is shown in Figure 15. The cellular phone is called Info Q here for short. The driving scenarios are

- (a) Before departure: i) Info Q Services; ii) Info GO Services; iii) Yellow Page Services.
- (b) On the move: i) Info GQ Services.
- (c) Soon arrives: i) Info GQ Services; ii) Yellow Page Services.
- (d) After arrives: i) Info Q Services; ii) Digital Home Services; and iii) Multimedia Services.



Figure 14 OBU is installed on vehicle.



Figure 15 Info Q is installed in cellular phone

(3) Design for Instant Information

At the part of instant information, we use the unified interface of traffic info center to obtain the real-time traffic info with broadcasting for every five minutes update. The OBU equipments can be cellular phone and OBU on vehicle use PUSH button to show ID number, Mobile number, GPS coordinate, request for service type, status report (accident, construction, traffic jam ...). This information (info) will send to TSP Platform. TSP will deliver related info to different division according to the service types.

CONCLUSION

In this paper, we will propose a reconfigurable framework on instant information query for advanced Telematics bundle services under heterogeneous wireless communicational environment. For instant information, we use the unified interface of traffic info center to obtain the real-time traffic info with broadcasting for every five minutes update. The OBU equipments can be cellular phone and OBU on vehicle use PUSH button to show ID number, Mobile number, GPS coordinate, request for service type, status report (accident, construction, traffic jam ...). We follow the .NET framework protocol, which looks like an extended Global Telematics Protocol (GTP 1.x) for mobile office service. We provide at least 18 bundle services together for subscribed members. The core functions of integrated bundle services can include four major services such as instant traffic info, driving info, information safety, and fleet management information, etc. However, they are not limited to those items.

ACKNOWLEDGMENT

CHTTL and THI would thank to IOT, MOTC to give us the chance to execute the project of Telematics. The main purpose of this three-year project is to support the instant information for northern scientific parks in Taiwan highway and support the World cup at Kaoshiung on 2009.

REFERENCES

- [1] Choi, J.-W., Han, W.-Y., Kim, C.-S., Kwon, O.-C., (2005/06). "Open Telematics Services Deployment on the gateway and Framework independent on Mobile Networks," Proc. of the international conference on wireless networks, pp. 374-379, Las Vegas, USA.
- [2].GTP 1.0 Specification (2003), <http://www.gtp.org>.
- [3]. Han, W.Y., Kwon, O.-C., Park, J.-H., Kang, J.-H. (2005/02), "A gateway and Framework for Interoperable Telematics Systems Independent on Mobile Networks," ETRI Journal, vol. 27, no. 1, pp. 106-109, Feb., 2005.
- [4] .NET Framework, <http://www.microsoft.com>.
- [5]. Kim, C.-S., Kim, J. and Kwon, O.-C. (2005/11). "Telematics Transport Gateway for Telematics Systems Independent on Mobile Networks," Proc. of the ITS World Congress 2005, San Francisco, USA.
- [6]. Kim C.S., Kim, J.-I., Han, W.-Y., and Kwon, O.-C. (2006/02). "Development of open Telematics Service Based on Gateway and Framework," ICACT 2006, p. 13491352.
- [7]. OSGi Framework Scope (2008), wikipedia website, <http://en.wikipedia.org/wiki/OSGi>.
- [8]. Pinart, C., Lequerica, I., and Barona, I. (03/2008). DRIVE: a reconfigurable testbed for advanced vehicular services and communications, in Proceedings of the 1st Workshop on experimental evaluation and deployment experiences on vehicle networks (WEDEV'08), pp.1-18, Innsbruck, Austria.
- [9]. Seepold, R. Madrid, M. N., Gomez-Escalonilla, J.-S. and Nieves, A.R. (2009). An Embedded Software Platform for Distributed Automotive Environment Management, EURASIP, Journal on Embedded Systems vol. Article ID 856962. pp.1-10.
- [10]. Tenqchen, S., Huang, Y.-K., Huang, H.-H., Chang, F.-S., Chen, K.-Y., Y.-H. Chen (2008). "Design of Open Telematics Service Based on Open Gateway and Framework: A Service Bundle Approach to Provide an Open Telematics Environment," Proceedings of 15TH World Congress on ITS, Nov. 16-2.
- [11] Yang, H.-H., Chien, C.-Y., Huang, P.-C. (YCH2009). OSGi Technology, ISMP.